TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

VILLE HÄMÄLÄINEN

USABILITY TESTING METHODOLOGY OF PROACTIVE HMIs FOR VIRTUAL CONTROL ROOM

Master's thesis

Examiners:
Prof. J. L. Martinez Lastra
Dr. A. Dvoryanchikova

Examiners and topic approved by the Faculty Council of the Faculty of Engineering Sciences on 9th of October 2013

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Degree Programme in Automation Engineering
HÄMÄLÄINEN, VILLE: Usability Testing Methodology of Proactive HMIs for Virtual Control Room
Master of Science Thesis, 58 pages, 3 Appendix pages
February 2014
Major subject: Factory Automation
Examiners: Prof. J. L. Martinez Lastra, Dr. A. Dvoryanchikova
Keywords: Usability testing, Proactive HMI, Virtual Control Room, User-Centered Design

The aim of this thesis is to develop methodology for usability testing of proactive HMIs. Usability is an important aspect of functionality of any Human-Machine Interface (HMI) since users directly interact with interfaces. Usability testing is, therefore, an important part of the development of HMIs. Emerging technologies, such as mobile devices and context-aware systems, pose new challenges for HMI developers due to new features and functionality. Since context of use is essential for HMI usability, HMI usability testing becomes especially important in context-aware systems. Both, the user and the HMI, need to have the same interpretation of context of use to ensure the usability of the HMI. At the same time, the context interpretation of a system should not conflict with the common sense of a user. A proactive HMI can be run on top of a context-aware system and it tries to predict next feasible action based on the context.

There are two challenges for testing usability in this kind of HMI. Firstly, the proactive HMI is a new technology and there are no previous studies made on proactive HMI usability testing. It has been conjectured that new features like proactivity, adaptivity and multimodality may challenge the results of measurements for earlier deffined attributes of usability and return the misleading results. Therefore, the usability testing methodology should be further elaborated in order to capture the usability of the proactive features. Secondly, the HMI is ment to run on mobile devices. Mobile devices have many constrains, such as limited computational resources, connectivity issues and varying display resolutions. Despite of these constrains, information should be accessible at any place and any time with mobile devices.

This thesis proposes a usability testing methodology of proactive HMI developed for Virtual Control Room (VCR). VCR is a proactive content-managing context-aware system; which was developed to increase the usability of embedded systems and human decision-making in data intensive environment. The methodology proposed in this thesis is based on Human-Centered Design (HCD) principles and consists of two stages: alpha (α) and beta (β) stage. In the α stage, traditional usability testing techniques are used with the addition of questions focused on getting information about what kind of proactive functionality the users would like to have on the HMI. In the β stage, a comparison of two different versions of the HMI, static and proactive, is conducted. This was done in order to test that the proactive functionality works well. It was found that

the standard attributes of usability are not enough to capture the usability of the proactive HMIs: thus the rates for efficiency (which is measured as a function of time) were low despite of positive verbal feedbacks from the participants. Assuming the importance of context of use, a new usability attribute – transparency – was created to capture usability parameters related to proactivity. The transparency was measured via the participants' reports on how easier they understand the HMIs elements and are capable to link them with a goal.

In order to illustrate the methodology in use, two use cases were selected to develop the usability testing methodology. In the first one, an HMI was being developed for Building Management (BM) domain. The HMI was developed for an automated five building complex to assist the living conditions in the buildings. In total this encompassed 25 apartments that were controlled via the VCR system. Another use case was an HMI that was being developed for Production Management (PM) domain. The HMI was generic software to monitor and control production line systems.

The approach selected in this thesis, allows an efficient way to receive feedback from usability tests and use this feedback to improve the proactive HMI. It is expected that the methodology developed in this thesis can be used to test other proactive HMI developed for automation systems.

CONTENTS

FIGURES

TABLES

ABBREVATIONS

AR	Augmented Reality
BM	Building Management
FAST Lab	Factory Automation Systems and Technologies Laboratory
GUI	Graphical User Interface
HCD	Human-Centered Design
HCI	Human Computer Interaction
HMI	Human Machine Interface
IR	Infrared
KPI	Key Performance Indicator
OS	Operating System
PM	Production Management
QR	Quick Response
SUS	System Usability Scale
TUT	Tampere University of Technology
UCD	User-Centered Design
VCR	Virtual Control Room

1. INTRODUCTION

Human-Machine Interfaces (HMIs) have been evolving fast in the past few years with the development of new mobile technologies and new devices such as smartphones and tablet computers. At the same time, quality of the HMIs is considered to be important. One of the key features of quality is usability. Therefore, usability testing becomes important on the HMIs. Additionally, new technologies, such as proactive functionality, are being developed. Proactive HMI uses context information to predict next feasible action. The aim of this thesis is to develop a usability testing methodology of proactive HMIs that are developed for Virtual Control Room (VCR).

Context-aware systems know the situation of the context. Since context of use is an attribute of usability (SFS-EN ISO 1998), the usability of the HMI becomes especially important. Both, the user and the HMI, need to have same interpretation of the context or usability will suffer. Examples of problems that might arise from the different understading of the context are that the user does not understand alarm, the user reacts incorrectly to alarm, safety or security issue goes unnoticed or user efficiency is lowered. Proactive HMI uses information from the context-aware system and tries to predict next feasible action based on it.

Traditionally visual presentation of a control room HMI is not good. VCR concept aims to make the information more readable, accessible even by novice users and mobile. Usability testing in mobile devices is still new research field and extensive studies on them have not been made. Mobile devices have restrictions which need to be taken into account conducting usability tests with them.

The rest of this chapter contains description of the problem that this thesis aims to solve and introduces the scope and outline of this thesis.

1.1. Background

Usability is an important aspect of the Human Machine Interface (HMI) design evaluation. The cost of maintenance and training can be greatly reduced by properly testing usability and of course the usability will get better (Lodhi 2010). Testing usability requires well thought methods which need to be adapted to the HMI in use. Because technologies behind HMIs change in time new methodologies for testing usability need to be developed.

Looking at the bigger picture, usability is a part of Human Computer Interaction (HCI) discipline. HCI emerged during the 1980s along with the invention of Graphical User Interfaces (GUI) (MacKenzie 2013). During those times research was done in the quality, effectiveness and efficiency of the interface. The research was mainly focused on how quickly and accurately can people do common tasks with GUI compared to text based interface.

Since then the HCI field has been growing steadily throughout the years and it has developed along with new technologies. The techniques for gathering usability information have gone through an evolution as well. Today there are standards of usability and well known techniques for getting information about usability of an HMI. The techniques include user testing, heuristic evaluation, observation, questionnaires and interviews (Nielsen 1993). In this work only user testing is focused from all of the techniques. User testing was chosen because it is well documented that user testing can expose many of the problems with the HMI. Additionally user testing requires as low as one person conducting the tests. Compared to, for example, heuristic evaluation, where many usability experts are required, user testing is easy to conduct. User testing is often referred and also from now on in this thesis as usability testing. Usability testing is the activity where users are given the product and they are asked to perform a series of tasks on the HMI.

Usability testing can be either formative or summative. Formative testing is usually done during the development of an HMI. Summative testing is done for a finished HMI. No matter what kind of testing is done a methodology needs to be developed. With weak, flawed or without methodology there is no science (MacKenzie 2013). The methodology in usability testing typically includes, among other things, participants, devices used and task scenario descriptions.

Existing standards for usability testing provide guidelines in a high level of generality. The existing methods for usability testing have been proven to be informative for desktop and web environments testing. However, the development of mobile HMIs and wanted functionality related to proactivity and adaptability of HMIs poses a new challenge. Usability testing methodologies for proactive mobile HMIs have not been created before.

1.2. Problem statement

There are two challenges for usability testing of proactive HMIs. First challenge is that what kind of information should the proactive HMI provide to the user. Proactive HMIs are very context dependent and they try to predict next feasible actions based on user action or context change (Boeck et al. 2007).

Usability tests evaluate user acceptance of the system with different criteria: system behaviour, user understanding of the information in the HMI, ease of use, etc. Context-aware systems expose different behaviours depending on the situation. It is important to develop usability testing methodology covering distinctive aspects of context-aware systems, and thus ensuring safety and comfortable use of the system.

The user should be in control of the HMI at all times. Proactive HMI reacts to changes in the context and information to the user is provided automatically. Following UCD guidelines helps to keep the user in control (see section 2.1.2). Conducting usability tests during development of the HMI and iterating the design based on the results helps to ensure that unwelcomed activity by the proactive decision making is discovered and redesigned.

The second challenge is that HMI for VCR is ment to provide mobility for the user as the HMI is run on mobile devices. Mobile devices should provide detailed information at anywhere and at any time, despite of constrains like limited resources, connectivity issues and varying display resolutions (Nayebi et al. 2012). The constrains affect usability testing and therefore testing needs to be planned well.

1.3. Research description

The aim of this work is to develop methodology for usability testing of proactive HMIs that are developed for VCR. This was done by applying principles of User-Centered Design (UCD), existing standards and guidelines needed for usability testing are modified and enriched with additional features comfortable to capture the usability parameters related to proactivity. Usability testing was chosen as the research method because the usability of proactive functionality needs to be tested on end users of the system. Usability tests were carried out in two different domains: Building Management (BM) and Production Management (PM). In both of the domains specific proactive HMIs for VCR were being developed. The domains resemble in functionality since they both have several user groups which require different information from the HMI, for instance alarms and warnings. Usability testing on proactive HMIs was made in both of the domains and results of the tests were analysed and compared.

The usability tests were carried out in two stages, alpha (α) and beta (β) stage. The α-stage was carried out with volunteers (16) and β-stage was done with the end users (23). The test protocol was developed for α stage. Based on the α-stage results the protocol was refined for the β stage test with end users.

1.4. Scope

This thesis focuses on usability testing on proactive mobile HMI which are run on data intensive environments for alarm management. The methodology proposed in this thesis is tested in two domains, building automation and production automation domains. The main purpose of the methodology is to gather usability information about the proactive features.

1.5. Thesis outline

The rest of this thesis is organized as follows: Chapter 2 presents literature review of proactive and mobile HMIs, user-centered design, usability and usability testing. Chapter 3 introduces the chosen use cases. In chapter 4 methodological approach is presented and the results from the usability tests are presented in chapter 5. Chapter 6 contains discussion about the results and lastly chapter 7 presents conclusions of the thesis.

2. LITERATURE REVIEW

This thesis focuses on usability of proactive HMI and usability testing. Usability is part of Human Computer Interaction (HCI) field. HCI field emerged along with the invention of Graphical User Interface (GUI) in the early 1980s and has been growing alongside the technology advanvacements in computer science. The experiments in the early stages of HCI research focused on questions such as is a certain task quicker and more accurate to do in GUI or in text-based command-line interface or the same question for two variations in a GUI implementation. Comparing different interfaces is still valid today in HCI research (MacKenzie 2013).

New technologies pose new challenges for HCI research. Although the basic questions in HCI research might stay the same, there is a need to improve the methodologies it is done with. The biggest challenge is to develop ways to test new technology features properly.

In this chapter, state of the art technology related to proactive HMIs and VCR is introduced. Challenges in HCI research for those technologies are brought up. Additionally, the chapter defines HMI usability as it is seen in this thesis and literature of usability testing is reviewed.

2.1. State of the Art Technology review

This section provides information about state of the art technology related to proactive HMI on VCRs. Additionally UCD is introduced.

2.1.1. Context-Aware Systems

The objective of context-awareness is that the right thing is given to the user at the right time and at the right way (Hofer et al. 2002). This means that the system needs to be aware of the context. From the user point of view, context is all the relevant information about the environment of the system. Context can be categorized into four different categories (Yılmaz & Erdur 2012):

- Physical context (e.g. light level, noise level, temperature)
- User context (e.g. user state, social situation, location)
- Computing context (e.g. network status, computing resources)
- Time (e.g. time of the day, year, season)

Context-aware systems use the context for their advantage. When the context changes the context-aware system should react to that change. A simple example would be that in smartphones the screen brightness can be set to automatic mode.When the light level detected by the device changes so does the brightness of the screen.

Instead of just using current context of use, the system can also utilize context history. Context history is simply a collection of the system past context and users' actions. It has many possibilities to improve the services the system can offer (Hong et al. 2009). For example, if a person in smart home environment, which a house with sensor and actuators attached to it, watches news always at 21:00 the system can use the context history and put the TV automatically on at 21:00 every evening. Context-aware system can also separate relevant and irrelevant information (Yılmaz & Erdur 2012). Taking the previous example further to context-aware system, the system could also know the days when the news are not shown and thereby not open the TV on those days.

2.1.2. User-Centered Design

User-Centered Design (UCD), also known as Human-Centered Design (HCD), is "an approach to systems design and development that aims to make interactive systems more usable by focusing on the use of the system and applying human factors/ergonomics and usability knowledge and techniques" according to (SFS-EN ISO 9241-210 2010). The benefits of using UCD are both economic and social. Highly usable products tend to be profitable. In consumer products users are willing to pay premium for well-designed products. Systems designed with UCD methods improve quality, for example, by:

- increasing the productivity of users and the operational efficiency of organizations;
- being easier to understand and use, thus reducing training and support costs;
- increasing usability for people with a wider range of capabilities and thus increasing accessibility;
- improving user experience;
- reducing discomfort and stress;
- providing a competitive advantage, for example by improving brand image;
- contributing towards sustainability objectives.

ISO 9241-210 provides a framework for UCD. It does not assume any particular design process but it fits well with the design of proactive HMIs. In proactive HMIs user centric design should follow the principles listed below:

- the design is based upon an explicit understanding of users, tasks and environments;
- users are involved throughout design and development;
- the design is driven and refined by user-centred evaluation;
- the process is iterative;
- the design addresses the whole user experience;
- the design team includes multidisciplinary skills and perspectives.

When designing an HMI four linked user centered design activities should be followed:

- understanding and specifying the context of use;
- specifying the user requirements;
- producing design solutions;
- evaluating the design.

In this thesis understanding and specifying context of use need to be considered from both the users' perspective but also from the perspective of the proactive HMI because the proactive HMI is also aware of the context. User requirements need to be specified for each user group separately. Differences between devices should be taken into account when producing design solutions for mobile devices. For example, HMI designed for tablet computer might need to be completely redesigned for smartphones since the screen size is a lot smaller. (SFS-EN ISO 9241-210 2010)

The ISO standard definition of UCD is quite formal and theoretical. More practical approach to user centered design is described in (Abras et al. 2004). It defines UCD as "a broad term to describe design processes in which end-users influence how a design takes shape". In (Abras et al. 2004) different UCD methods are described. In this thesis we focus only on usability testing which is one of the UCD methods.

There are great benefits using user centered design but there are some downfalls as well. Interviewing users and conducting usability tests with them obviously takes time and money. On top of this, user centered design teams should include persons from different disciplines such as psychologists, sociologists or anthropologists to get the best of it (Abras et al. 2004). Some teams or companies just do not have that opportunity. Table 2-4 shows the advatages and disadvantages of UCD.

Table 2-1: *Advantages and disadvantages of UCD* (Abras et al. 2004).

Advantages	**Disadvantages**
Products are more efficient, effective, and safe	It is more costly.
Assists in managing users' expectations and levels of satisfaction with the product.	It takes more time.
Users develop a sense of ownership for the product	May require the involvement of additional design team members (i. e. ethnographers, usability experts) and wide range of stakeholders
Products require less redesign and integrate into the environment more quickly	May be difficult to translate some types of data into design
The collaborative process generated more creative design solutions to problems.	The product may be too specific for more general use, thus not readily transferable to other clients; thus more costly

2.1.3. Proactive HMI

The word *proactivity* refers to anticipatory, change-oriented and self-initiated behaviour. This means that proactive systems are monitoring change and anticipating what the user needs. In HMIs this means that it predicts next feasible action based on context change or user action. Proactive HMIs are all the time monitoring the current context and can suggest the next step from context or from the history of the user (Boeck et al. 2007). For getting the proper context proactive HMI need to monitor the world around it which requires sensors and actuators (Tennenhouse 2000).

Proactive HMI requires to be connected to context-aware system in order to obtain the information that is needed. The context-aware system can either be on the same device as the proactive HMI or connected by a network. Proactive HMI and context-aware system connection can be seen as a client-server relationship. The context-aware system is the server which feeds the client, the proactive HMI in this case, with information. The user interacts only with the proactive HMI. The proactive HMI reacts accordingly to situations (by performing actions or informing user) and interacts with user in most natural way by providing him information based on the current context (Chen et al. 2008).

The challenge in developing a proactive HMI is to have the HMI react to context changes as the user would expect it to react. Problems might arise in case the context is understood differently by the HMI and the user or the HMI reaction to context change is different what was expected by the user. The problems can be such as user does not understand alarm, user reacts incorrectly to alarm, safety or security issue goes unnoticed or user efficiency is lowered. Another challenge is to know what to monitor.

It is important to know when proactive functionality is appropriate and when it is not approriate (Vico et al. 2011). For example, if the user is in hurry he or she can have different expectations from the proactive functionality than when he or she is not in a hurry and monitoring user “time pressure” is important. Following user-centered design guidelines helps to deal with the challenges in proactive HMI development (see section 2.1.2).

2.1.4. Mobile HMI

Mobility has become increasingly popular in the HMIs over the last decade and many new devices are developed every year. Most popular mobile devices today are smartphones and tablet computers but new devices are coming to the market as well. For example, a major smartphone and tablet producer Samsung has published a new smart watch (Samsung 2013) this year and Google is developing Google Glass (Google, Google Glass 2013). Both of the new devices are wearable computers and highly mobile. Mobile HMIs need to be developed and tested even though the devices might change throughout the years.

Mobile devices have some key features which make them different from other devices. Mobile devices are used wirelessly, they have small screen sizes and different screen resolutions (Inostroza et al. 2012; Nayebi et al. 2012). Mobile devices are also used usually only for short periods of time. The restrictions need to be taken into consideration when testing usability on the devices. For example, with small screen sizes the information might not be as visible as on a larger screen and buttons need to be big enough.

Current mobile devices such as smartphones and tablet computers are capable to run most programs. Due to this some HMIs have been directly ported, or with just minor modifications, from desktop computers to mobile devices without considering usability of the HMI. Although mobile devices can run many desktop HMIs, the HMIs on mobile devices are used differently.

A big constraint with current mobile devices, and especially smartphones, is that entering text is quite difficult. Especially for older people, it is hard to touch the right letters with such small buttons. In the design this should be always taken into consideration and the need for entering text should be minimized. For example, instead of giving an option to write a message in the HMI there could be an option to choose from predefined messages to send. In some cases it is unavoidable however, for example when log in is required. (Wisniewski 2011)

Today some of the major mobile operating system (OS) developing companies have been enforcing their own design rules for mobile applications. For example, Google (Google, Android Design 2013) has a design guide for Android applications. The guide includes such things as navigation in applications, icon styles and touch feedback. Many of the items in the design guide are made to keep the applications consistent with each other and the platform, but at the same time, it also gives tips on how to make applications unique from each other and how to improve usability of the application. The guidelines also tell how they should be considered during the development and testing of applications. Another big company today in the mobile OS market is Apple with its iOS. Apple also has guidelines for designing mobile applications and they review applications submitted for the App Store based on characteristics such as interaction with multi-touch screen, device orientation changes and gesture control (Nayebi et al. 2012).

A good way of starting development of multi-device software is to start with the mobile application first (Wisniewski 2011).

When starting development and design for the device with various constrains, such as small screen size, they need to be addressed from the beginning. It is easier to move from a constrained device to other devices than vice versa. For instance, with this design philosophy development could be started on a smartphone and moved there to tablet computers and desktop computers.

Well designed mobile HMIs are especially attractive for older people because they are used with touch-panels.

In (Petrie & Bevan 2009) pointing time was compared between a touch-panel and mouse controlled HMI. In the study pointing time in the mouse controlled HMI increased with the age of the participants but stayed the same with the touch-panel HMI. The study also states that learnability is better for touch based interfaces. If text input is minimized or removed completely mobile HMIs are very useful for older people.

2.1.5. Alarm management

Alarm management is an important part of any automation system. Alarm management system detects abnormal values or equipment states in the automation system and reports them to the operators. The operator of the system is responsible for taking action on the alarms. Typically many alarms are generated per day depending on the domain and size of the facility. For example, in a process plant there can be one alarm every two minutes and even up to 100 alarms in 10 minutes if something abnormal has happened (Liu et al. 2003). Another example is that hundreds of alarms can be generated during one day in Microsoft's Puged Sound campus (Smith 2012). The amount of alarms will strain the alarm management system operators' attention and some important alarms might get unnoticed. Therefore, there is a need for intelligent alarm management systems.

In alarm management system there has to be a definition of what is an alarm, priorities for alarms, what are the normal performance values in the system and what are the limits, and how the alarms are reported (Berwanger 2013). Alarm management system needs fault detection and diagnostics (FDD) to work. FDD consists of two steps. First step is where all the data is collected from devices and sensors, and the second step is where this data is analysed and abnormal values are detected (Smith 2012). The alarm management system then takes the abnormal values and decides if the user should be notified about them.

Applying proactive solutions for alarm management systems is a must in future systems since the data amount increases every year with a staggering pace. This means that number of produced alarms will increase as well. Key challenge in alarm management is to recognize the alarm priority and how other events are related to it (Smith 2012). With proactive functionality alarms can be prioritized and grouped based on various parameters, such as the user role, user state, other active alarms and alarm type.

2.2. HMI Usability

There are several definitions of usability in the literature and the definition varies based on the field of research. In this chapter usability definitions related to HMIs are investigated.

In (Nielsen 1993) a practical definition of usability is given and usability is defined through five different attributes of usability. The definitions can be seen in Table 2-2.

Table 2-2: *Definition of usability attributes by* (Nielsen 1993).

Attribute	Definition

Learnability	The system should be easy to learn so that the user can rapidly start getting some work done with the system.
Efficiency	The system should be efficient to use, so that once the user has learned the system, a high level of productivity is possible.
Memorability	The system should be easy to remember, so that the casual user is able to return to the system after some period of not having used it, without having to learn everything all over again.
Errors	The system should have a low error rate, so that users make few errors during the use of the system, and so that if they do make errors they can easily recover from the. Further, catastrophic errors must not occur.
Satisfaction	The system should be pleasant to use, so that users are subjectively satisfied when using it; they like it.

Nielsen brings up an important criteria for usability. Learnability and memorability are important today because HMIs hold great amounts of data. The challenge is how to present the data to the user in a way that he can learn to use the HMI fast and he can return to it even after a long period of not using it without having to re-learn everything in it.

Low error rate should be a given to any HMI. Errors in HMI affect to all the other usability criteria as well. It is really important to remove all errors that are detected during usability testing.

From the ergonomics for visual display terminals point of view the most general definition of usability can be found in the ISO 9241-11 standard. It defines usability as: “Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” (SFS-EN ISO 9241-11 1998). The standard also defines the following usability attributes that can be seen in Table 2-3.

Table 2-3: *ISO 9241-11 definitions of usability attributes* (SFS-EN ISO 9241-11 1998).

Attribute	**Definition**
Effectiveness	Accuracy and completeness with which users achieve specified goals.
Efficiency	Resources expended in relation to the accuracy and

	completeness with which users achieve goals.
Satisfaction	Freedom from discomfort, and positive attitudes towards the use of the product.

Nielsens definition of learnability can be seen as part of efficiency in the ISO 9241-11 standard. Learnability is efficiency in the beginning of use. Nielsens definition of memorability can be seen as part of effectiveness in the ISO 9241-11 standard. Memorability is effectiveness when some time has past when the product was last time used.

Similar to the previous definitions of usability can also be found in (Quesenbery 2001). The wording is a little different but the meaning of the attributes stays the same. In (Quesenbery 2001) engaging is the only term not found in the previous definitions. It brings up how the interface should be visually attractive and mentions satisfaction. Ultimately, the attribute engaging is similar to satisfaction in the other definitions or it atleast results in good satisfaction.

Efficiency is important evaluation criteria in this thesis. HMI needs to be efficient for making it usefull for the users. Efficiency can be grouped with learnability because learnability is efficiency in the beginning of use. In usability testing efficiency can be measured with time on task.

Effectiveness is an important usability criteria because effectiveness considers how well tasks can be done with the HMI. Errors in the HMI affect directly effectiveness, the lower the error rate the better the effectiveness of the HMI. Effectiveness can be measured in usability test with completion rate, error rate and assist rate. Memorability is a special case of effectiveness. It is the effectiveness when the user has not been using the HMI for a while and he or she needs to remember how to use it.

Satisfaction is also an important criteria for usability. Satisfied users keep using the HMI and are less likely to change it another HMI provided by a different company. Satisfaction is how the user feels about using the HMI. Satisfaction can be measured with questionnaires focusing on satisfaction on use of the HMI. Table 2-4 shows comparison of different definitions of usability attributes introduced in this chapter.

Table 2-4: *Different definitions of usability attributes compared to each other* (SFS-EN ISO 9241-11 1998; Nielsen 1993; Quesenbery 2001).

ISO 9241-11	**Nielsen**	**Quesenbery**
Efficiency	Efficiency, Learnability	Efficient, Easy to learn
Effectiveness	Memorability, Errors/Safety	Effective, Error tolerant
Satisfaction	Satisfaction	Engaging

2.3. Usability testing

The term usability testing in literature often refers to any technique to evaluate a product or system. Usability testing in this thesis refers only to the activity where users are given an HMI and they are asked to perform a series of tasks on the HMI. This activity is sometimes also refererred as user testing. Usability testing can be divided into two general types (Barnum 2011):

- Formative usability testing – done on developed HMIs and testing goals focus on diagnosing and fixing problems. Typically small studies and repeated during development.
- Summative usability testing – done on finished HMIs and the goal is to establish baseline of metrics or validating that the HMI meets requirements. Generally requires larger numbers for statistical validity.

In this paper we focus on formative usability testing. Usability testing can also be divided based on the method used on the test. Different methods are, for example, lab experiment, field study or remote test. Regardless of what type of testing is done usability testing can usually be divided into three different phases. The phases include (Barnum 2011):

- planning of the usability test,
- conducting the usability test,
- and analysing results of the test.

In the next sections the three phases are explained in detail.

2.3.1. Planning usability test

Planning phase of usability test is very important because it determines how the rest of the test is done. A test plan should always be written. Several items need to be addressed in the test plan. The items include:

- **Establishing test purpose and goals –** Purpose is a high level reasoning for performing the test. The purpose can be, for example, that a new feature has been developed and it needs to be tested on end-users. Based on the purpose a list of goals should be made. Because of time and money constraints all of the goals might not be able to be tested so they should be ordered by priority. Evaluation criteria can be used to help create the list.
- **Participant characteristics and recruiting –** First thing to do is to determine the characteristics of the target participants for the test and the amount of participants. Usually there are subgroups of users with different characteristics. It should be determined which and how many subgroups are wanted to participate the test. In less formal usability test research has shown that four to five participants find 80 percent of the usability deficiencies but for statistical significance minimum of 10 to 12 participants is required. When the characteristics and amount has been determined the participants can be recruited.

- **Determining participant incentintive if any –** Incenvite might help in recruiting participants but it should be made feel like a thank-you gift and not a bribe. Incentive should not influence the remarks the participants make during the test. It can be anything from movie tickets to cash and the amount depends on the participants.
- **Determining what to test and creating list of tasks for the test –** What to test depends on what you want to learn in the test. The test goals give high level information about what is wanted to learn but they need to be further divided into smaller pieces. When it is known what to test a list of tasks can be produced. The task list contains tasks that the parcipant will do in the test. Good tasks scenarios should feel real tasks to the participants. If it is made that way the participants will do a better job at trying to complete them.
- **Desiding on test environment, equipment and logistics –** The environment that the test tries to simulate should be described and equipment that is needed for the test need to be listed.
- **Set dates for the test and deliverables –** Dates should be set for the deliverables and testing. Generally, four or five one-hour sessions is a good amount for one work day. Table 2-5 shows answers for test times from usability experts.
- **Determine quantitative and qualitative feedback methods –** Typically getting qualitative feedback is focused more on formative usability tests and quantitative in summative testing. Quantitative data can be divided into two subgroups: performance and preference data. Performance data can be, for example, time on task or number of errors on task. Preference data is gathered from user opinions or thought process and includes, for example, responses to questions and questionnaires. Qualitative data, on the other hand, can be gathered throughout the test from observations of the participant. It can be, for example, comments that the participant made or observations done by the testing persons about the participant expressions or body language.

Table 2-5: *Usability experts answers for questions about usability testing times based on 30 answers.* (Barnum 2011)

Question	Mean	Median	Mode	Low	High
How many one-hour sessions are you good for in a day?	4.87	4.75	4	3.5	6.5
How long does it take to analyze 10 one-hour sessions and write up the results?	39.8 hours	40 hours	40 hours	6 hours	80 hours
How many years have you been doing user testing?	13	14		2	29

After the test plan has been written, test times have been determined and participants have been recruited, the next step is to actually do the test. The next section describes how the test can be conducted.(Barnum 2011; Rubin & Chisnell 2008)

2.3.2. Conducting usability test

Before conducting the test, a pilot test should be done on one or two participants. Pilot test is made to test that the protocol works and to determine the time spent to conduct one test. If there are any problems detected during the pilot test they can still be fixed before the actual test.

When the pilot test is done the actual testing can begin. Typical usability testing is a "one-on-one" situation where the test moderator giving the participant tasks and observing the participant conducting the tasks. There can also be more people in the testing team, for example, a logger to take notes, observers which can be for instance developers, or technician to help with the technical things.

Usability testing contains a lot of important things that should be remembered to do during the test. For this reason, it is recommended to do a checklist in order to not forget anything important. Different checklists can be made for different parts of the test. There can be, for example, a checklist for week before the test, one day before the test, for the day of the test before any testing is done and checklist used for after each participant. The checklist before the test should contain items that make sure you are fully prepared for testing. The items can be, for example, check that the video equipment is set up and ready, and check that the software and hardware are working. Usually small things get forgotten so it can contain also items such as check that there is enough battery for the laptop.

When everything is ready the testing can begin. Usability testing typically contains the following parts:

- **Introduction** – In introduction the participant is briefed what the test is for, what he needs to do, what is expected from him and anything else that is considered important for him or her to know. During the introduction necessary forms are signed (e.g. video consent form).
- **Pre-test questionnaire** – Pre-test questionnaire is made to get background information about the participant. For example, previous experience with similar HMIs can be asked.
- **Tasks –** In the task scenarios the participant will do series of tasks with the HMI and the test team observers how he or she manages to do them. Required performance values should be recorded and participant expressions.
- **Post-test questionnaire –** Typically made to rate the overall experience and satisfaction using the HMI. The questions should be linked to the goals of the test.

- **Debriefing –** Ending the test and expressing gratitude to the participant for doing the test. Participant can also be asked if he or she would like to participate in other tests in the future.

When all the participants have made the test there should be a lot of material to analyse. How to analyse the findings is described in the next section. (Barnum 2011; Rubin & Chisnell 2008)

2.3.3. Analysing results of usability test

After the test, the gathered data needs to be compiled, summarized and analysed. Usually the data from the usability tests is analysed in two stages. The first stage is shortly after the test where highlights of the test should be analysed. The first analysis is made for getting the most obvious issues on paper when the test is still fresh in the memory. Preliminary report can be made from the first analysis. This allows the developers and designers to start working on the discovered issues. Since the report is provided shortly after the test it should be made clear that it is just a preliminary report and the report items might change for the final report.

After the preliminary report work on a final comprehensive report can be started. The final report contains detailed findings and thorough analysis of the test. First the data needs to be compiled. This means that the raw data needs to be put into a format that makes it easy to understand. There are many ways to compile data.It can be done, for example, by using lists, matrices, storyboards and so on. The main point of compiling data is to make it understandable and start discovering patterns. After the data is compiled it needs to be summarized. Summarizing data helps to see the overall picture, for example, where the participants made the most errors and where they managed well.

After the data has been compiled and summarized it needs to be analysed. In the analysis phase the test objectives should be taken into account and data should be analysed based on them. For example, if there was two versions of the product a comparison between the versions should be made. Usability test result analysis can be divided into two categories, qualitative data analysis and quantitative data analysis.

- Qualitative data analysis
- Quantitative data analysis

In the next chapters the two different analyses are explained.

2.3.3.1 Qualitative data analysis

Qualitative data analysis measures user opinions and behaviour. It is used to describe and explain social phenomena (Pope et al. 2000). Typically qualitative data analysis is time consuming and requires expert knowledge to get valid results.

In usability test, qualitative data is typically gathered with “think aloud” method where the participants describe their thoughts while conducting tasks. Qualitative data analysis of the comments may give some insight why the participants made certain errors or why they were slow at a certain task. Typically, usability tests include a set of open questions after the participant has finished the tasks. The questions should be based on test objectives to give right information that can be analysed with qualitative mehods.

2.3.3.2 Quantative data analysis

Quantitative data is the data that can be put into numbers. In usability test, it can be, for example, time on task, success/completion rates, errors (and recovery), failure (failure of whole task), assists (moderator assists or using help within the system, questionnaire results or anything else that can be quantified. After counting the values that are wanted they need to be analysed statistically. (Barnum 2011; Rubin & Chisnell 2008)

There is perception in statistical analysis that the sample size should be large, typically above 30, but it is usually impossible to get that many participants when doing usability testing. The cost of time and money would be just too much. Usability testing sample sizes can go as low as one or two. Luckily, there are ways to deal with small sample sizes. One way is to calculate confidence interval for the results which make the results more meaningful. The information how to calculate confidence interval is discussed in section 2.3.6.2. Additionally choosing to use numbers instead of percentages helps. Instead of using 67%, 2 out of 3 can be used to make the result more meaningful for a reader. (Sauro & Lewis 2012; Barnum 2011)

In any usability test it is important that the participants are “representatives” of the user base. It does not matter what the size of the sample is if the participants do not represent the user base. Without this, there is no logical basis of generalizing the results for the whole user base. Instead, the results would just apply those people involved in the test. Ideally, the sample should be randomly selected from the representative group, but usually this is not possible. The most important is to get people who represent the user base. (Sauro & Lewis 2012)

2.3.4. Laboratory vs. Field Usability Testing on Mobile Devices

Usability can be tested with field usability testing or laboratory testing. Since context of use is important for mobile devices, it should be simulated in laboratory usability testing. In (Kjeldskov et al. 2004) a study was made to compare laboratory and field usability tests for mobile devices. The authors concluded that testing in laboratory setting can be more efficient. More usability problems can be discovered when the context of use is simulated.

If the development, and especially the usability testing, is run on a tight budjet paper prototyping might be a good idea. With paper prototyping usability issues can be discovered with low cost. Usability issues can be discovered with paper prototyping before they are implemented. However, paper prototyping is not sufficient to discover usability issues of the final product and especially interaction or performance issues (Kangas &

Kinnunen 2005). Testing should also be done in laboratory when running a tight budjet since field testing is more expensive (Kangas & Kinnunen 2005).

Study comparing laboratory and field usability testing was made in (Kaikkonen et al. 2005). In the study, 20 users made the same test in laboratory environment and another 20 users made it in field environment. The results from the study tell that same usability issues can be discovered in both, field and laboratory environment. Field test also takes longer. Setup for the whole test and setup between participants take longer time resulting in the whole test taking twice as long as in laboratory. Field study can be better when user behaviour and environment needs to beexamined but, for discovering usability issues in the system, laboratory testing is more efficient.

Field testing needs to be prepared better since there are more variables and distractions in the field (Kaikkonen et al. 2005). Long time usability experts might handle distractions well but, for new usability tester, the distractions and variables can cause the test to fail. For new usability testers, there is a need for clear structure in the test and field test distractions can cause them to make mistakes. A novice usability practitioner can have difficulty handling something that was not planned for the test.

2.3.5. Limitations of usability testing

Usability testing has its limitations. Although usability testing is widely considered useful technique, it cannot guarantee that the usability of the HMI is good. Usability testing can never tell with 100 percent accuracy that the HMI will be usable even with a large study. Reasons for this are that (Rubin & Chisnell 2008):

- **Usability testing is every time an artificial situation.** Testing always affects the results of the test no matter what kind of testing is done and what techniques are used. The very act of conducting the test can affects the results.
- **Test results do not prove that the HMI works.** Even large statistics from usability test do not prove that the HMI works. Statistical significance is not a guarantee, it is a measure of the probability that the results were not due to chance.
- **Participants are rarely fully representative of the target population.** Participants are only as representative as one’s ability to understand the potential end-users of the HMI. In some cases, they might be actually known for the time of the test but who is to say they do not change in the future.
- **Usability testing is not always the best technique to use.** In some cases usability testing is not the most effective, cost efficient or accurate way to test and other techniques such as heuristic evaluation might be better. It should be considered depending on the product and development phase which technique should be used.

Although usability testing has limitations, it is considered very effective way of finding usability issues from HMI and improving its usability when done correctly. Conducting usability tests minimizes risks of releasing an unstable or unusable HMI.

2.3.6. Tools for usability testing

There are different tools and guidelines to help with usability testing. In the following sections usability testing tools which were used in this work are described.

2.3.6.1 SUS questionnaire

SUS questionnaire is a quick and easy way to evaluate usability of any system. It was developed in response to the need of having a quick questionnaire at the end of usability testing when the participant has gone through series of tasks and might be very frustrated and would not finish a long questionnaire. SUS questionnaire is a simple ten point questionnaire providing global view of subjective assessment of usability (Brooke 1996). It is a Likert scale where answers are given on 5 point scale ranging from "strongly agree" to "strongly disagree" and the questions alter between negative and positive items, i.e. in half of them common response is "strongly agree" and in the other half "strongly disagree".

The participants taking the questionnaire should be instructed to answer the questions with what comes to mind at first and not think about the questions for too long. All the questions should be answered and, if the participant cannot respond to some of the questions, they should be marked as the center of the scale. The questions and Likert scale can be seen in appendix A.

The SUS questionnaire was developed to give a global view of subjective assessments of usability (Brooke 1996). Although it is a short questionnaire, it covers a variety of aspects of HMI usability, such as the need for support, training, and complexity. Thus, has a high level of validity for measuring usability of a HMI.

2.3.6.2 Calculating confidence interval

In usability testing there is almost never an access to the entire user population. Instead, we have to make estimates from the results that we have and how they would translate to the people that were not tested. This is why confidence interval is required. Confidence interval is the range of values that one thinks will have a specified chance of containing the unkown population i.e. the population that the test did not cover (Sauro & Lewis 2012). For example, if a person was asked how long it takes to eat lunch. Even his or her best answer (for example 30 minutes) would be wrong by few minutes or seconds. Therefore, it is better to give a time interval as an answer, such as 25 to 35 minutes. The same applies to usability test result statistics. In usability test statistics confidence interval use margin of error and it is actually twice the marging of error (Sauro & Lewis 2012). Confidence interval for usability test results gives information about value of the result and its precision. Appendix B explains how confidence interval is calculated.

3. USE CASES DEFINITION

Two use cases were selected to develop the usability testing methodology. The use cases were taken from BM and PM domains. In both of the use cases proactive HMI for VCR was being developed. In the BM use case, the HMI was developed for a automated building with assisted living in Tampere Finland. In PM use case, the developed HMI was a generic software for monitoring and control of a production line environment. In section 3.1 the VCR concept is described, in section 3.2 intelligent alarm management for the both use cases is explained, and in sections 3.3 and 3.4 detailed description of the use cases is given and in section

3.1. Virtual Control Room

VCR is a proactive content-managing context-aware system and it aims to facilitate human-machine interaction and human decision making in data intensive environment (Nieto Lee et al. 2013). With VCR data is pre-processed and filtered to reduce information overload to the user. With VCR, users can monitor system state at any location and at any time in a comfortable manner.

Current automation systems provide centralized monitoring systems which are not intelligent or user friendly. It is often that people communicate via text messages or even radio about the system conditions. System alarms are hardcoded and many of them tend to trigger at the same time, which is overwhelming to the system operator and hard to pinpoint where the problem is exactly. This results in incorrect diagnostics, long maintenance times or even safety issues.

With VCR, traditional control room interfaces can be presented virtually and an interface for the VCR can be accessed any where and at any time. But most importantly, VCR provides more intelligence to the interaction between the user and the system. VCR is aware of the context and it provides information only when and where it is needed. Alarms are not just triggered individually but they are grouped and provided only to the users that need them. Additionally, guidance and tips are provided with the alarms. With VCR, the user can carry an HMI with him at all times.

In both BM and PM domains VCR is utilized. In both domains there is a large amount of information produced by the system which needs to be processed. VCR concept allows effective information filtering and provides information to the right users at the right time. This reduces the overload of information to the user of the system. In both domains user roles are important part of the systems since different users require different information from the system. VCR provides information to only the user roles that require it.

3.2. Intelligent alarm management

VCR is aiming to provide intelligent alarm management. In this thesis, alarms were divided into three different severities: alarm, warning and notification. Alarms are divided into categories based on the time limit for the operator to take an action on the alarm and consequences of the alarm. The alarm severity categorization can be seen in Table 3-1. With this categorization, a user can easily understand the severity of an alarm and react correctly. The user's attention can be focused on the most important alarms at the right time. It is important that the operator does not have any doubt which is the highest priority alarm in the HMI.

Table 3-1: *Alarm categorization.*

Name	Time limit for operator action	Consequences
Alarm	Requires immediate action from the user.	Severe financial loss or severe safety consequences.
Warning	Requires action within some time limit.	Moderate financial loss or moderate safety consequences.
Notification	Requires action from the user but there is no time limit.	Minor financial loss or minor safety related consequences.

The highest priority alarm can be, for example, a fire alarm which requires immediate action from the operator. The consequences of a fire alarm are life threatening and financial losses can be substantial. Warnings can be, for example, equipment failures which cause parts of a production line to stop or produce lower quality products. This kind of warning might cause moderate financial loss to the company. Notifications can be, for example, situations where quality of products have dropped over a period of time. This requires either a maintenance person or production engineer to look into the matter at some point.

In this thesis the alarms are grouped in the VCR model in both domains. VCR provides context-aware proactive information filtering for the alarms. Alarms can be grouped based on user role, other alarms and user preference. The VCR model tries to prevent overloading the users' attention with information. Grouping of alarms supports this effectively.

3.3. Building management

The chosen use case for developing usability test methodology on BM domain is automation system which is controlling and monitoring living conditions inside the buildings. The system automates five buildings with twenty five apartments. Each apartment consists of one room and has wide range of sensors: door state sensors, IR sensors in bathroom and the room, temperature sensor, water leak sensor, fire alarm sensors, air ventilation sensors, etc. The system reacts to the current context by producing correspondent actions and notifications to the users.

In total, the system has more than 1000 data points from the apartment sensors, common areas, and technical rooms, which are monitored by users of the system. Since the system's technical solution provides web-based interfaces, it is possible for the users to do partial monitoring of living conditions with usage of smartphones and tablets along with PC and laptops.

3.3.1. User requirements

User roles description and understanding is an important criteria for usability test's methodology, because each user role has its own requirements and a way to interact with the system. There are three different user roles for the BM use case: inhabitants, maintenance and nurse personnel. Each of the user roles require particular kind of information. They have different kinds of tasks that they need to accomplish with the HMI. The user role descriptions and their main HMI requirements, formed after the field studies, can be found in table 1.

Table 3-2: *BM use case user roles*

User roles	Tasks	Information needs	Notes	Main Mobile HMI Requirements	Display Devices
Inhabitant	Own apartment monitoring and control (e.g. lights and doors). Contacting nurse or maintenance personnel. Receiving alarms (e.g. fire alarm). Receiving notifications (e.g. notification about weather change)	Own apartment status (e.g. light and door statuses)	Might have disabilities	HMIs as simple as possible	Smartphone
Maintenance	Buildings monitoring and control (e.g. temperature set point limits for the apartments). Receiving alarms and warnings (e.g. water leak warning). Receiving notifications (e.g. notification about weather change).	Alarm and warning statuses. Status of embedded devices. Status of the whole building site. Setting values for the embedded devices. Contact information of the nurses.	Mobility important Multiple tasks at the same time Some tasks need to be done fast	HMIs should be informative, but not overwhelmning with information. HMIs should not require any prior knowledge for usage.	Tablet computer, laptop, desktop computer.

Nurse	Inhabitants health monitoring. Sending messages to inhabitants. Receiving alarms (e.g. alarm about inhabitant health condition). Receiving notifications (e.g. notification about weather change).	Alarm and warning statuses. Inhabitant health status. Contact information of maintenance personnel.	Short time to use the system		

Inhabitant's information needs of the system are restricted to their own apartment. They require such information as the door statuses of their apartment and control for lights. The inhabitants will use the system on smartphone. Since the inhabitants might have some disabilities, like low visual acuity, it is important to provide them with a simple HMI.

The maintenance personnel use the system on tablet computers and laptops. They require information about all the apartments and specific information for discovering faults and fixing them. Alarms and notifications are therefore very important for maintenance personnel. They need to be understood correctly, come timely, and possibly give instructions on what to do.

Nurse personnel also use the system on tablet computers and laptops and they require mostly information about the inhabitants' health status. They also need to know schedules of inhabitants and other workers. On top of this, they do not have much time to use the system.

Many of the tasks and information needs are related to notifications and alarms. Therefore, usability methodology focus should be put on testing the system in different conditions, allowing user to evaluate different aspects of the system functions and provide sufficient and substantial feedback to system developers.

3.4. Production management

The chosen use case for developing usability test methodology for production management domain is automation system for control and monitoring of production line. The production line used was FASTORY line which is situated in TUT Finland and it was originally fully automated production line made for assembly of mobile phones. Today the industrial equipment is outdated but the production line has been integrated with the newest information technologies. It is a testbed for research, development, and demo activities. FASTORY has the main components of a production line: robots, transportation system, tools, end effectors, raw material, working stations, loading and unloading stations, and a buffer station. All the assets can be used to recreate a large amount of control and maintenance scenarios that can happen during the operation of existing assembly lines.

FASTORY can be used as a part of the HMI usability testing to create adaptive user interfaces which provide information to different personnel based on their state, skills, preferences, tasks and the current state of the system. FASTORY gives the possibility to implement this concept for the common personnel that can be found in an automated production system: manager, supervisor, operator, and maintenance personnel, making it suitable to do the usability testing on HMI developed for it.

The HMI being developed for this use case is generic HMI for manufacturing environments. The HMI is developed for Android devices and especially for tablet computers. The HMI provides proactive functionality to the user. The HMI also provides support for the different user roles that can be found from the production line environment.

3.4.1. User requirements

User roles description and understanding is an important criteria for usability tests methodology, because each user roles of the system has its own requirements and a way to interact with the system. User roles in PM domain are also important for the usability test methodology. Different user roles have different requirements for the HMI and its functionality. There are five different user roles for this use case: management, supervision, engineers, shop floor personnel and maintenance. Each of the user roles require different kind of information. They have different kinds of tasks that they need to accomplish with the HMI. The user roles, actors that fill the user role, and description can be found in Table 3-3.

Table 3-3: *PM use case user roles.*

User roles	**Actor**	**Description**
Management	Production Engineer Manager	Supervision and management of production technicians.
Supervision	Line Manager	Supervision and management of specific production line.
Engineers	Production engineer	Increase the efficiency of the processes by using new technologies, installing new equipment or modifying current machines.
Shop floor personnel	Team Leader	Person capable to replace any line operator and fix simple problems.
	Line Operator	Assembly of certain parts in production; providing raw materials to robots and removing completed products; identification of problems.
	Material	Responsible for providing material to operators in

	Supplier	the assembly line.
Maintenance	Maintenance Technician	Maintenance technician is on charge of fixing hardware and software problems in the robotic cell. Its an outsource element that is contacted when problems in the automated assembly line are detected.

Each of the roles has their own requirements for the HMI and HMI needs to adapt to the requirements. Management personnel require high level information such as key performance indicators (KPI). Additionally, the information might need to be gathered from a long time period. Supervisors only require most of the time information for their own production line. Engineers require information about the efficiency of the production lines and where the bottlenecks are. Shop floor personnel require real time information for production lines and material situtations. Maintenance personnel require information for errors and faults and how to fix them. These were examples of what information each user role requires. There is also common functionality between all the user roles such as events (warnings, alarms, etc.) or contacting other personnel.

In the usability testing different user roles and their requirements for the HMI were be taken into account. Key actors from the user roles were identified for the usability tests and they were line manager, line operator and maintenance technician. They were line manager, line operator and maintenance personnel. All the possible interaction with the HMI is represented with the key actors and all the required functionality can be tested with them. In Table 3-4 the required functional features by each actor can be seen.

Table 3-4: *PM HMI functional features required by the users highlighting the actors which participated in the usability test.*

Actor	**KPI values**	**System states**	**Tickets**	**Daily tasks**	**Material availability**	**Maintenance instructions**
Line Manager	X	X	X			
Line Operator		X	X	X	X	
Maintenance Technician		X	X			X
Production Engineer Manager	X					
Production engineer	X					
Team Leader		X	X	X		
Material Supplier					X	

4. METHODOLOGICAL APPROACH

The methodological approach presented in this work is based on principles of UCD, existing standards and guidelines for usability testing and proactive functionality testing developed for this work. The principles of UCD can be seen in chapter 2.1.2. Existing standards and guidelines of HMI usability are discussed in chapter 2.2. Usability testing guidelines for proactive functionality is presented in the following sections.

The general protocol for usability testing in this methodology includes two stages: alpha (α) and beta (β) stage. In both stages formative usability tests were done on the developed HMIs. Between the stages the HMIs were refined based on the results of the α stage usability tests and normal development plans for the HMIs. The α stage focused on usability issues related to understanding the HMI and its functionalities such as navigating in the HMI and accessing information. The β stage focused on testing proactive functionality such as alarm grouping and functionality on user state changes.

4.1. Evaluation criteria of usability

4.1.1. Usability attributes

The evaluation criteria for the tests were chosen based on the preliminary research on developed HMIs and literature review. Preliminary research included usability expert's research and discussions about context-aware and proactive HMI usability evaluation. Based on the preliminary research and literature review guidelines for validating and evaluating usability in proactive HMIs were developed. The evaluation criteria presented in this thesis take into account those guidelines and the literature review made in this thesis (see section 2).

In this methodological approach it is proposed a definition of usability for proactive HMIs, which is extending (SFS-EN ISO 1998), as: "*Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" which is transparent to the user and the system.* It is stated that usability for proactive HMI can be defined through four different usability attributes: **Effiency**, **Transparency**, **Effectiveness** and **Satisfaction**. The new definition is based on the literature review in chapter 2 and the new definition compared to other definitions can be seen in Table 4-1.

Table 4-1: *Usability definitions by ISO 9241-11, B. Sheiderman, J. Nielsen and W. Quesenbery* (Quesenbery 2001; SFS-EN ISO 9241-11 1998; Nielsen 1993)

New definition	ISO 9241-11	Nielsen	Quesenbery

Efficiency	Efficiency	Efficiency, Learnability	Efficient, Easy to learn
Effectiveness	Effectiveness	Memorability, Errors/Safety	Effective, Error tolerant
Satisfaction	Satisfaction	Satisfaction	Engaging
Trasparency	-	-	-

Efficiency means that tasks can be done in timely manner with the HMI. Efficiency is an important attribute in any HMI but especially important when there are safety critical tasks to be done with the HMI. For example, managing alarms requires good efficiency from the HMI.

Effectiveness means that tasks can be done accurately. Effectiveness is an important attribute in proactive HMI. The user needs to stay in control and do tasks how he wants but at the same time receive help from the proactive functionality.

Satisfaction means that the user is satisfied using the HMI and does not feel discomfort when using the HMI. Good satisfaction means that the user would be likely to want to use the HMI again after using it.

The existing usability attributes were not enough to capture the proactive functionality in HMIs and that is why a new attribute is introduced in this work. Context-aware system requires the same understanding of context from the system and the user. **Transparency** is a new attribute introduced in this thesis. It can be seen as: *Interpretation of the context should be coherent between the user and the HMI*. If the context of an intelligent system is misinterpretated by a user it decreases the usability of an HMI.

4.1.2. Usability attributes evaluation

Efficiency was evaluated through time on task. Time on task is calculated by how long it takes for the participant to complete a task. Time on task was taken on the tasks that had completion parameter set to them, i.e. they could be completed. A task that can be completed is for example: “What is the temperature in the apartment currently.”

Effectiveness was evaluated through three different means: tasks completion rate, error rate and assist rate. Most of the tasks had binary completion, i.e. they were either completed or not. Completion rate could then be calculated easily by dividing the completed tasks with total amount of tasks that could be completed. Error rate is calculated by dividing error count with total amount of tasks. An error was an incorrect action taken by the participant during a task. Assist rate was calculated by dividing assist count with total amount of tasks. An assist was an action taken by the moderator to help the participant to complete a task. Effectiveness was evaluated on all the tasks that had binary completion.

Satisfaction was evaluated in the post-test interview and in two post-test questionnaires. In the post-test interview a question asking for overall impressions of the design of the HMI evaluated satisfaction. The first questionnaire was a SUS questionnaire (see section 2.3.6.1) and the second was semi-closed satisfaction questionnaire. Both of them evaluated satisfaction.

Transparency was evaluated in tasks and questions. Transparency was tested by making a user go through different elements of the HMI and asking if they are as the participant expected them to be. Each test started with a look and feel task where participants were asked to explain what they saw in the starting screen and if they understood everything on it. Participants were also asked about icons and if they were easy to understand. In other tasks transparency was evaluated with questions like: “What kind of messages would you like to be able to send to inhabitants?” Additionally, in proactive events made by the HMI, the user was asked if the HMI gives appropriate information about the event.

The test included post-test interview questions and post-test questionnaire questions which aimed to get insights from the user expectations for the HMI. In open interview questions transparency related questions were:

- Was there anything unclear to you?
- Name three things you would change in the HMI.

With the questions unclear elements from all the used screens could be discovered. Some of the weaknesses were already discovered when going through the tasks but the questions gave confirmation to the weaknesses. In questionnaires, there were also some questions which focused on evaluating transparency in the HMIs. The questions were:

- Icons are clear.
- Colors related to the sensors were clear.
- Sensor statuses were clear.
- I found the system unnecessarily complex (SUS question 2).
- I needed to learn a lot of things before I could get going with this system (SUS question 10).

4.2. Test setting

This chapter explains where the tests were made and what kind of environment was used. In this methodology, laboratory usability testing was chosen as the method of testing. Laboratory testing was chosen because it is more efficient than field testing and same usability issues can be found in laboratory test as in field test (Kjeldskov et al. 2004; Kaikkonen et al. 2005). Additionally, laboratory testing has less external variables and distractions which make it feasible even for novice usability tester.

4.2.1. Test setting in α stage

The α stage in both use cases were held in TUT premisis in conference room environment. Necessary equipment for usability testing and context simulation was brought to the room. The following section describes the test setting in α stage in detail.

4.2.1.1 Building management

Test time was one day and it was divided into five test sessions. Each session included one participant. The participants were scheduled to take one hour and there was a 15 minute break between each participant. The day had also 45 minute lunch break.

The test team consisted of a moderator and a logger. The moderator did all the interaction with the participants, giving the instructions to the test, tasks, post-test questionnaires and debriefing. The moderator also set up the test and took notes to a notepad during the test. The logger took notes silently on the test to an Excel spreadsheet which had letter codes for different actions (for example, participant made an error or new task started) and automatic time calculation.

The layout of the usability was a computer lab room at TUT in Fast (RL201) with long tables for several laptops and good lighting. The tablet computer was placed on a table and the participant was sitting on a chair while using the HMI. The moderator was sitting next to the participant and the logger was sitting on the other side of the participant. Both the moderator and the logger could see clearly what the participant was doing with the HMI.

Registration devices that might affect results were an Android application called Smart Voice Recorder (Smartmob Development 2013) which was used to record audio of the session. CamStudio version 2.7 was used to record the laptop HMI screen (Camstudio.org n.d.).

Type of devices that were used were Lenovo Thinkpad T410 laptop running windows 7 which was used for the laptop HMI. Samsung Galaxy S2 running Android version 4.1.2 “Jelly Bean” was used for the smartphone HMI.

Simulations done were all the data and events displayed on the HMI during the test. No real data from any smart building system were shown on the screen. The simulations were performed by a portable demoBox. The demoBox could be seen as attempt to scale real system to the small box size. It consists of one Inico S1000 controller (Inico Technologies Ltd 2013), which corresponds to real system controllers, and connected buttons, which represent the simulated sensors from environment. The box was connected to a laptop which was running the system gateway. Thus, the states of the simulated sensors were controlled via the demoBox in a similar way as in the real system. The tasks in the usability tests were selected in a manner that the demoBoxes could correspond entirely to the real system.

4.2.1.2 Prodution management

Test time for the test was three days and each testing day included four test sessions. The participants were scheduled to take one hour and there was a 15 minute break between each participant. Each day had also 45 minute lunch break.

The test team consisted of a moderator and a logger. The moderator did all the interaction with the participant, giving the instructions to the test, tasks, post-test questionnaires and debriefing. The moderator also set up the test and took notes to a notepad during the test. The logger took notes silently on the test to an Excel spreadsheet which had letter codes for different actions (for example participant made an error or new task started) and automatic time calculation.

The layout for the the usability was a conference room at TUT (Fast skyroom) with a table for 7 people and good lighting. The tablet computer was placed on the table and the participant was sitting on a chair while using the HMI. A video camera was placed above the tablet computer with tripod so that it filmed the screen from above. For the augmented reality part of the test, a QR code was placed on a wall. When the participant used the augmented reality part, he had to stand up and go to the wall with the tablet on his or her hand. A smartphone was used to film the screen on the augemented reality part. Pictures of the layout can be seen in Figure 4-1 and Figure 4-2.

Registration devices that might affect results were a standard video camera which was used to film the screen of the tablet computer. The camera was next to the participant on a tripod facing the screen from up to down. Additionally, Samsung Galaxy S2 smartphone was used to film the augmented reality part. Both, the moderator and the logger had standard laptops to take notes.

Type of devices that were used were Samsung Galaxy Tab 2 10.1 (Android version 4.0.1 “Ice Cream Sandwich”) which was used to run the tested HMI. QR code printed on a white A4 paper was used for the augmented reality part.

Simulations done were all the data and events displayed on the HMI during the test. No real data from any production line were shown on the screen.

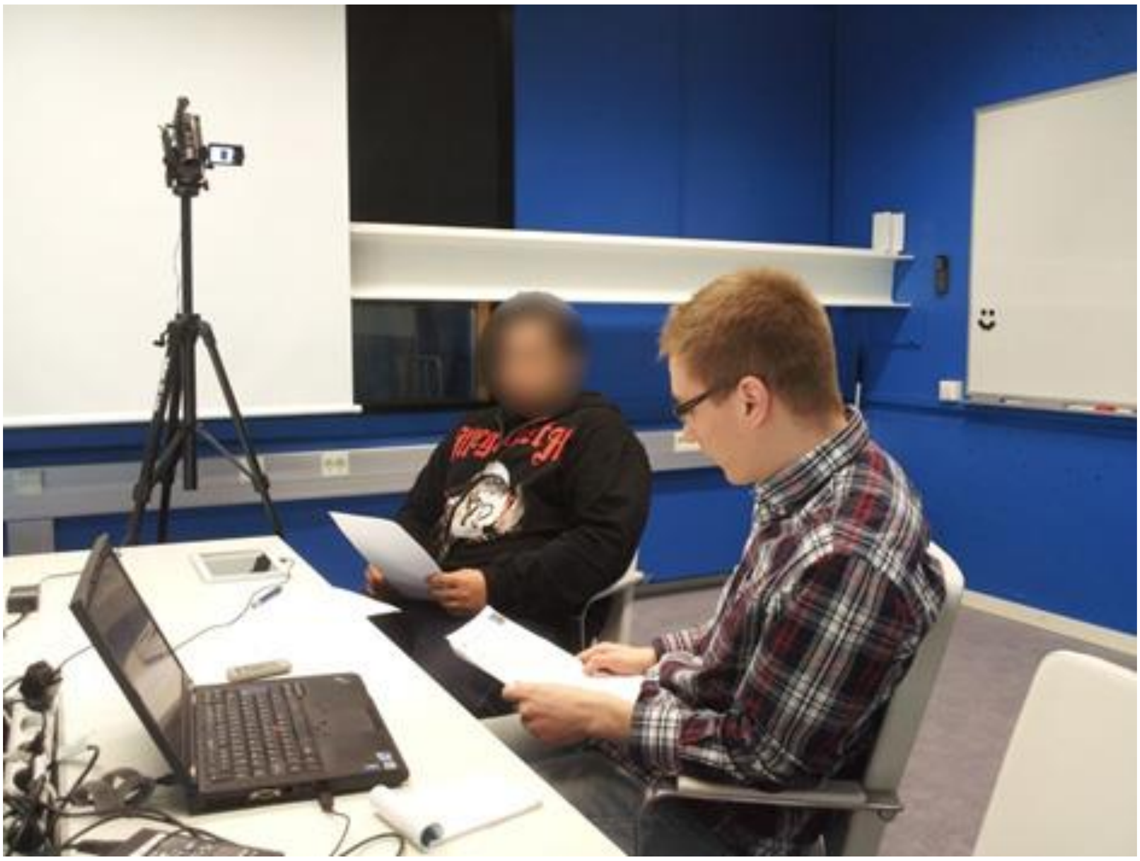

Figure 4-1: *PM α stage usability test setting where moderator is reading introduction to the test (the participant on the left and moderator on the right).*

Figure 4-2: *PM α stage usability test augmented reality part test setting where the participant is using the augmented reality part of the HMI.*

4.2.2. Test setting in β stage

The β stage tests were held in the end user facilities in a conference room environment. The rooms used are comparable to laboratory environment, and necessary equipment for usability testing and context simulation was brought with the usability test team. The following sections describe test setting in the β stage in detail.

4.2.2.1 Building management

Test time for the test was one day with 7 participants during the day. The participants were scheduled to take 30 minutes and there was 15 minutes break between each participant. The day had also 45 minute lunch break.

The test team consisted of only a moderator. The moderator did all the interaction with the participant, giving the instructions to the test, tasks, post-test questionnaires and debriefing. The moderator also set up the test and took notes to a notepad during the test.

The layout for the usability test was a conference room in building complex situated in Tampere, Finland. The room used had a large table with chairs and big windows giving good natural lighting. The laptop computer and smartphone were placed on a table and the participant was sitting on a chair while using the HMI. The moderator was sitting next to the participant. The moderator could see clearly what the participant was doing with the HMI.

Registration devices that might affect results were an Android application called Smart Voice Recorder (Smartmob Development 2013) which was used to record audio of the session. CamStudio version 2.7 was used to record the laptop HMI screen (Camstudio.org n.d.).

Type of devices that were used were Lenovo Thinkpad T410 laptop running windows 7 which was used for the laptop HMI. Samsung Galaxy S2, running Android version 4.1.2 “Jelly Bean”, was used for the smartphone HMI.

Simulations done were all the data and events displayed on the HMI during the test. No real data from any smart building system were shown on the screen. The simulations were performed by two portable demo boxes. The demo boxes could be seen as attempt to scale real system to the small box size. It consists of one Inico S1000 controller (Inico Technologies Ltd 2013), which corresponds to real system controllers, and connected buttons, which represent the simulated sensors from environment. The boxes were connected to a laptop which was running the system gateway. Thus, the states of the simulated sensors were controlled via the demoBoxes in a similar way as in the real system. The tasks were selected so that the demo boxes could correspond parts of the real system. Also part of the test was done with the usage of control panel running in the smartphone. This panel was used in the tests to create multiple alerts at the same time, when demo boxes buttons were not enough for triggering needed amount of different alarms and notifications. In the control panel, the buttons were related to the tasks in the usability test. With the touch of a button all the alarms and notifications needed for the task were triggered.

4.2.2.2 Production management

Test time for the test was three days with total of 16 sessions were made. The participants were scheduled to take one hour and there was 15 minutes break between each participant. Each day had also 45 minute lunch break.

The test team consisted of moderator, a logger. The moderator did all the interaction with the participant, giving the instructions to the test, tasks, post-test questionnaires and debriefing. The moderator also set up the test and took notes to a notepad during the test. The logger took notes silently on the test to an Excel spreadsheet which had letter codes for different actions (for example participant made an error or new task started) and automatic time calculation.

The layout for the usability was a conference room at FluidHouse Ltd. Finland with a large table and good lighting. The tablet computer was placed on the table and the participant was sitting on a chair while using the HMI. A video camera was placed above the tablet computer with tripod so that it filmed the screen from above. A stack of paper to represent material on the production line was placed on the table next to the participant. Pictures of the layout can be seen in Figure 4-3 and Figure 4-4.

Registration devices that might affect results were a standard video camera which was used to film the screen of the tablet computer. The camera was next to the participant on a tripod facing the screen from up to down. Both, the moderator and the logger had standard laptops to take notes.

Type of devices that were used were Samsung Galaxy Tab 2 10.1 (Android version 4.0.1 “Ice Cream Sandwich”) which was used to run the HMI being tested. Stack of paper with cover, which had a barcode, was used to represent material used in the production line.

Simulations done were all the data and events displayed on the HMI during the test. No real data from any production line were shown on the screen.

Figure 4-3: *PM β stage test setting where participant is doing a task in the test (moderator at back and participant in front).*

Figure 4-4: *PM β stage test setting where participant is filling post test questionnaire (logger at back and participant in front).*

4.3. General Protocol

The general protocol for the usability tests were divided in two stages: α and β stage. α stage focused on discovering typical usability issues and the β stage focused on proactive support testing. Detailed protocols from the two stages are in the following sections.

4.3.1. General protocol in α stage

The general protocol followed the same pattern in both use cases. The tests were performed individually. Each test included the following stages:

- standard introduction,
- pre-test questionnaire,
- task scenarios,
- post-test questionnaire,
- and debriefing.

The introduction was considered as an important part of the test protocol since it placed the HMI tester to the context of the system and daily goals of the system users, allowed him or her to familiarize with the testing environment, and considered the ethical agreements (e.g. the protection of the testers' personalities, and the restrictions for the record dissemination). With good introduction, the user could be put in the situation of the test and the test felt less artificial. It was underlined that in focus of the test is the HMIs design rather than skills or efficiency of a tester performance. The general test protocol was explained and, for instance, the participants were encouraged to comment aloud every interaction with the HMI.

The pre-test questionnaire consisted of short open questions dedicated to discover the previous experiences of working in similar environment and with similar devices, for example, previous experience with mobile devices was asked. This phase allowed to control the effects on the testing parameters like efficiency and satisfaction.

The main block of the test consisted of the task scenarios. The scenarios were introduced sequentially; the implementation of a task was started after the moderator's signal. The task scenarios were selected to prove the compliance of the evaluation criteria requirements. First task, which was similar in all the tests, "Exploration of the start screen" was about the first experience with the tested HMI – general clarity of the HMI and first impressions. After that, the tasks placed the participant to his or her role in the environment and he or she was asked to do various tasks with the HMI. Some tasks included, for example, interpretation of events (for instance, understanding the difference between a warning and an error event).

The task scenarios were made to represent situations in real use cases in order to make the user feel less like he is conducting a usability test. The scenarios were made to follow a story throughout all the scenarios. The simulation of the context was also made to make the test feel less artificial.

The post-test questionnaire consisted of three blocks:

- Open questions over the general impression on HMI (as example of question – What is your overall impression of the design),
- Standard System Usability Scale (SUS) questionnaire (Barnum 2011) with 10 closed statements to be ranked by five grades from “strongly disagree” to “strongly agree” (as example of a statement - “I think I would need technical support to use the HMI”), and
- Satisfaction semi-open statements with seven point likert scale and comment field (as example - “The system makes execution of my tasks more effective”).

With the questionnaires, statistical data was gathered and the users could also mention things that they did not realize to say during the tasks. The test was ended by debriefing with expression of gratitude for the participation.

4.3.2. General protocol in β stage

The HMIs were developed further between the α and β test stages. During the time in between, the issues found in α stage were dealt with. The β stage protocol was a little different than the α stage protocol. Each test in the β stage included the following stages (see previous section to get detailed description of the stages):

- standard introduction,
- pre-test questionnaire,
- proactive HMI task scenarios,
- proactive HMI post-test questionnaire,
- static HMI task scenarios,
- static HMI post-test questionnaire
- and debriefing.

The main objective of the β stage was to evaluate the proactive support of the HMI. There were specific tasks that were tested twice with different versions of the HMI. One version was proactive HMI and the other version was static HMI. The proactive version had all the proactive functionality developed for the HMI. In the static version, all the proactive functionality was removed. With the tasks proactive support could be evaluated when the results from the similar tasks in different versions were compared. Additionally, post-test questionnaire results could be compared from the different versions.

The test protocol followed within-subject method with the two versions of the HMIs which means that all the participants made the tasks with the proactive and static version. The participants were also divided by half, with one half starting with static version of the HMI and the other half starting with proactive version of the HMI.

Some of the tasks were not testing proactiveness of the HMI and therefore they were only done once. The tasks were similar to the α stage tasks. They were the ones that aimed to get certainty that end users understood everything on the HMI.

4.4. Participants

In the α stage, participants were recruited from TUT. In the β stage, participants were end-users of the product. Having end-users of the product made sure that the participants were representative in the usability tests. Overview of the participants in each test can be seen in Table 4-2.

Table 4-2: *Overview of participants in each test.*

Case	Stage	Participants	#	Age	Notes
BM	α	TUT engineering students	5	20-30	
BM	β	Inhabitants in the building complex	5	20-55	No previous experience of the HMI; Some had conditions like ADHD or panic attacks.
BM	β	Nurse personnel in the building complex	2	20-40	Experience from previous version of the HMI
PM	α	TUT engineering students	11	20-30	
PM	β	FluidHouse workers	16	20-55	8 operators, 5 supervisor, 3 maintenance personnel

4.4.1. Participants in α stage

Participants for the α stage of testing, in both uses cases, were recruited from TUT. Therefore, all of the participants in the α stage were engineering students.

For the BM use case, the participants were recruited from fellow master thesis workers from the Department of Production Engineering and in there from FAST Lab. The participants volunteered for the test. There were 5 participants which were all male and 20 to 30 years old. None of the participants had experience in working in an automated building or living in one. All of the participants had experience with using smartphone.

Participants for the α stage of PM test were university students from TUT. 11 participants were recruited and all of them were engineering students and aged between 20 to 30 years old. There were 7 male participants and 4 female participants. The participants were recruited from the university’s internal intranet website where a newspost was made both in Finnish and in English. In the newspost, it was told that the test will be made in English and participants are expected to have good English skills. It also said that participants will receive a 20€ gift card for the university book store.

The experience, to related field and devices, varied between the participants. Four of the participants had previous experience working in a production line in different job descriptions. Only one of the participants did not have previous experience with smartphones. Additionally, 8 of them had experience with tablet computers.

4.4.2. Participants in β stage

Participants for the β stage were end-users of the HMIs in both use cases. For the BM domain participants were inhabitants and nurse personnel in the automated building complex (see section 3.3). All of the BM test participants volunteered for the test. The inhabitants were living in the building complex and they had no previous experience with building automation HMIs. They used the smartphone HMI and only one of them had experience with smartphones. All of them said that they have some previous experience with computers (laptop or desktop computers). Additionally, some of the inhabitants had conditions like ADHD or panic attacks.

There were only two nurse personnel in the β test because there was no possibility to get more suitable participants in the BM use case. The nurses were workers in the building complex and they had experience with previous version of the HMI. One participant had worked 8 months in the building complex and the other one had worked there for 2 years. Both nurse participants mentioned that they use the current HMI daily. They used the HMI with laptop computer in the test.

Participants in the PM use case were workers in FluidHouse Ltd. Finland. There were participants from three different user groups:

- 8 operators
- 5 supervisors
- 3 maintenance personnel

All the participants volunteered for the test. The participants were selected because it allowed to test all the functional requirements of the different user roles. Most of the participants were operators because they are the biggest portion of production line personnel. They use many of the key features of the HMI. Most of the operator and maintenance participants did not have much experience with tablet computers or smartphones. Only 6 participants had experience with Android devices. The supervisors had more experience with tablet computers and smartphones. Many of the operators and maintenance personnel said that they do not have good English skills. This might have affected the test results as the HMI was in English. The participants were offered buns and biscuits for participating in the test.

5. RESULTS OF THE USABILITY ATTRIBUTES MEASUREMENTS

The results of the tests were analysed in two stages. Right after the test, the first impressions were discussed with the people conducting the test. Based on the discussion a list was made. The list included top positive findings, top negative findings and top surprises found by each person conducting the test. This enabled the team to report early findings right away and the development team could start working on some of the issues immediately. (Rubin & Chisnell 2008)

After the initial impressions, all the data gathered in the test was analysed and summarized in the final report. The data was analysed based on the evaluation criteria (see section 4.1). Summary of results from all the tests is presented in this chapter. The results are grouped by the evaluation criteria used in the tests.

5.1.1. Efficiency

Efficiency was evaluated through task times. In BM β stage test proactivity was tested on the laptop HMI. Testing was done with two different versions of the HMI, static and proactive version, to determine if the proactiveness worked well. The proactive version had alarm grouping functionality which ment that some of the sensor which indicated the same alarms were grouped together and only one alarm was shown. The user could then drill down into details to find out which sensor had caused the alarm. Additionally, alarms could be grouped based on the building or apartment they came from.

The times for the tasks are shown in Table 5-1. As it can be seen the results from the proactive HMI is signifigantly better. The time on task decreased for every task in proactive HMI compared to static HMI. The participants were also asked to compare the two different versions of the HMI and they clearly stated that the proactive version was better and it gave more information in a more convenient way. For instance, one of the participants commented on the proactive HMI that: “The system tells well where a problem is and what kind it is.”

Table 5-1: *BM β laptop HMI task times comparison.*

Task	**Static HMI average time (sec)**	**Proactive HMI average time (sec)**	**Decrease in proactive HMI (sec)**
Explaining the nurse alarm	90	27	-63
Explaining the alarms	60	45	-15
Finding highest	53	20	-33

priority alarm			
Putting the alarms in order by importance	78	63	-15

In the PM β test the tasks for the different user groups were different and the task times are divided into three different groups based on the participants. The participants were operators, supervisors and maintenance personnel. In the PM β stage testing was also done with two different versions of the HMI: static and proactive version.

In Table 5-2 can be seen a comparison between times on tasks for the static and proactive version on the operator tasks. The task time for log in was quite similar in both versions. Exploring error event was faster in proactive version. This shows that the quick options for the error work efficiently. The exploring instruction and sending information to material handler tasks were faster in the static version but in the proactive version the quality of the information received or sent was higher. Additionally, one of the operator participants commented the proactive HMI that: "The instructions make work easier."

Table 5-2: *PM HMI operator task times comparison.*

Task	Static HMI (sec)	Proactive HMI (sec)	Difference (sec)
Logging in to the application	85	89	+5
Exploring operator instructions	177	272	+96
Exploring error event as operator	363	324	-39
Sending information to the material handler	111	158	+47

In Table 5-3 task times comparison can be seen for the supervisor participants. The proactive version was clearly faster in all the tasks. All the tasks for the supervisor were similar in both versions but the proactiveness in the proactive version helped the user to do the tasks faster. One of the supervisor participants commented that: "The production line status can be seen clearly from the starting screen which helps my work."

Table 5-3: *PM HMI supervisor task times comparison.*

Task	Static HMI (sec)	Proactive HMI (sec)	Difference (sec)
Logging in to the application	96	60	-37
Exploring supervisor instructions	208	167	-41
Exploring error event as supervisor	261	174	-87
Receiving multiple notifications at the same time	158	50	-107

In Table 5-4 task time comparison can be seen for maintenance participants. Log in was quite similar in both versions. Exploring error event was faster in proactive version which again indicates that the quick options for the error work efficiently in the proactive version. Exploring instructions was faster in static version but the information was multimodal in the proactive version.

Table 5-4: *PM HMI maintenance task times comparison.*

Task	**Static HMI (sec)**	**Proactive HMI (sec)**	**Difference (sec)**
Logging in to the application	109	101	-8
Exploring error event as maintenance person	280	201	-79
Exploring maintenance instructions	174	247	+73

Overall the proactive version was clearly more efficient for the PM HMI in the β stage test. Especially exploring the error event stands out because all three user groups finished the task faster with the proactive version. This shows that proactive functionality for the error event works efficiently.

5.1.2. Effectiveness

Effectiveness was evaluated through tasks completion, assist and error rate. For the BM laptop HMI, the assist and error rates were higher and completion rate lower in the static version HMI. This indicates that the laptop HMI proactive version is more effective. The results can be seen in Table 5-5.

Table 5-5: *BM laptop HMI effectiveness statistics.*

Domain	**Test stage**	**Device**	**Completion rate**	**Assist rate**	**Error rate**
BM	β proactive	laptop	1	0.11	0.56
BM	β static	laptop	0.88	0.25	0.75

The results for the PM β stage can be seen in Table 5-6. Some of the participants did not have experience with smartphones or tablet computers. Also, the HMI language was English and some of the participants did not have good English skills. Because of the issues the error rate is high for the static version. In the proactive version the error and assist rates are clearly lower despite the issues. This indicates that the proactive functionality results in less errors and the proactive HMI is clearly more effective.

Table 5-6: *PM HMI effectiveness statistics.*

Domain	**Test stage**	**Completion rate**	**Assist rate**	**Error rate**
PM	β proactive	0.95	0.41	1.12
PM	β static	0.92	0.69	2.31

5.1.3. Satisfaction

Satisfaction was evaluated in post-test interview and questionnaires. For the BM laptop HMI, the participants mentioned in the open question that the HMI was good, professional looking and that it was easy to use and understand. For the smartphone HMI, the participants said in both stages that the HMI was very simple and easy to use. In the β stage, the participants were end-users of the HMI and they even mentioned that they would like to have the HMI for themselves.

The post-test questionnaire for satisfaction was composed to determine satisfaction from all the parts of the HMIs. The satisfaction questionnaires consisted of seven questions with a scale from 1 to 7. In BM HMIs satisfaction questionnaires the participants were overall quite satisfied with the HMIs. All of the scores are above average score (4), which indicates that the users were satisfied with the HMI. There was no statistical difference between the results from the α and β stages.

In PM HMI satisfaction questionnaire, the results were overall quite good as well. In the α stage question 3, the system does not contain too much information, was the only question to get a little lower score than the other questions. This was probably due to the fact that some of the information in the HMI was not correct because the HMI was still in development. Additionally, many of the participants did not have experience in working in a production line, therefore, it might be hard for them to know what information should be available on the screen.

In the PM β stage proactive HMI the satisfaction questionnaire, the only question that got a little lower score than other questions in the β stage was question 6: I feel in complete command of the system. The participants mentioned in that question, for example, that they gave low score because of the language or because they did not get to use the system for too long. There is no statistical difference between the results from the α and β stage but overall all of the scores are above the average which indicates that the satisfaction for the HMI was good.

In the PM β stage, comparison between proactive and static satisfaction questionnaires were made. The first four questionaire questions that evaluated the satisfaction were asked separately for both static and proactive version of the HMI in order to compare the results. There is no statistical difference between the results from the different versions.

5.1.4. Transparency

Transparency was evaluated in tasks that measured user understanding of the HMI elements and in post-task interview and questionnaires with the especially introduced questions. In BM domain, look and feel task was especially important for the smartphone HMI since the inhabitants had never used it before. Nurse and maintenance personnel had experience from the previous version of the laptop HMI. In the α stage it was discovered that some of the terms used in the HMI were not clearly understood by the participants. The terms were changed after the α stage and in the β stage the terms were understood better. In BM interview, answers gave confirmation to weaknesses discovered earlier such as terms used, icon looks, clickable elements and naming of categories.

In PM domain, look and feel tasks were also made. In the α stage, some of the PM domain terms were not understood since the participants were engineering students and not all of them had experience in working in a production line environment. The terms were explained to them since they were important to understand in the rest of the tasks and that way unnecessary errors were eliminated because of not understanding those terms in other tasks.

In PM α stage test, it was discovered that the error, warning and notification screen information was not understandable and should be changed. The problem was that the screen did not give enough information about the event and it was difficult to determine what had happened. Additionally it was discovered that it is important to indicate which elements on the screen are clickable.

In PM β stage, one of the main issues with transparency was that the language of the HMI was English instead of the local language (Finnish). Based on the results, it was suggested that the language is better to have in native language of the user. Again in the β stage some of the terms were not understood by the participants. These kinds of terms are different in different companies and it was suggested that the terms should be determined when the HMI is customized for a specific company.

The results from the post task questionnaire questions are gathered to Table 5-7 and Table 5-8. The questions in the BM use case gave high scores. This indicates that transparency was good in the HMI. In the PM HMI, the scores are lower for the β stage. This was most likely because the participants in the β stage were working in a production line environment and they already had an idea how things should be. They did not, for example, like if there were terms that they did not understand in the HMI.

Table 5-7: *Transparency questionnaire results in BM domain.*

Question	**Stage**	**BM laptop HMI**	**BM smartphone HMI**	**Scale**
Icons are clear.	α	5.4	6.4	1-7
	β	7.0	6.0	1-7
Sensor statuses were clear.	α	5.0	6.4	1-7
	β	7.0	6.6	1-7
I found the system unnecessarily complex.	α	75	100	0-100
	β	100	85	0-100
I needed to learn a lot of things before I could get going with this system.	α	100	95	0-100
	β	100	85	0-100

Table 5-8: *Transparency questionnaire results in PM domain.*

Question	**Stage**	**PM HMI**	**Scale**
Icons are clear.	α	4.9	1-7
	β	4.8	1-7
I found the system unnecessarily complex.	α	82	0-100
	β	72	0-100
I needed to learn a lot of things before I could get going with this system.	α	80	0-100
	β	56	0-100

6. DISCUSSION

The usability testing methodology proposed in this thesis is aiming to gather information about the proactiveness of the HMI. This was done by having two different versions of the HMI, static and proactive, and comparing the results between the two versions. In the results for efficiency most of the tasks were made clearly faster with the proactive version. In some of the tasks the time was faster for the static version but the results are not so straightforward in those cases. Time lost is usually interpreted as a bad thing but not necessarily in proactive HMI. The time lost in proactive HMI can also mean that there is more information available to the user or the quality of the information is better.

In the PM instructions task for operators and maintenance personnel, the static version was faster because in the proactive version the information given to the user was multimodal. The participants had to listen to the instruction read aloud and it took longer time than just reading them. It ment that it took longer to listen to the instruction but it is also better in some cases when you can not look at the screen. The sending information to the material handler task for operators was faster in the static version but in the proactive version a picture of the material was send with the message and therefore the quality of the information is better in the proactive version.

Efficiency alone may not work for proactive HMI usability testing but another usability attribute is required. Transparency can used alongside efficiency to determine if the tasks are made efficienctly but also that the user has the right information.

The satisfaction questionnaire results are unfortunately quite close to each other and when comparing results from α and β stages or between proactive and static HMI there is no statistical difference. This was mainly due to the fact that the sample sizes were quite small. It can be said though that all of the satisfaction questionnaire results are above average which indicates that the satisfaction in the HMIs was good. Additionally in both of the use cases the participants made positive comments on the HMIs. On the BM HMIs the participants commented, for instance, that they would like to have the HMI for themselves (inhabitants) or that the HMI was improvement from the previous version (nurses). On the PM HMI the participants especially commented positively on the proactive HMI saying, for instance, that the route on the map was very good and that the instructions would make work easier.

The main result from the transparency evaluation was information about how important the vocabulary is in the HMI. In the most common level vocabulary means that the language used in the HMI is good to be the local language. In more specific level the vocabulary needs to be specific to the company, factory or work place. Different companies can use different terms for the same things. In the PM use case the vocabulary was taken from a different factory than where the end users worked and it was discovered that some of the terms were not understood by the users.

Additionally when evaluating transparency it was discovered that icons are important in the HMIs and how they behave when clicked. It is important to indicate in the HMI which elements are clickable and provide feedback to the user when they are clicked. Icons need to look like what they represent and be appropriately sized on the screen.

7. CONCLUSION

Contributions

This thesis proposes a methodology for usability testing of proactive mobile HMIs that were developed for Virtual Control Room. In order to capture the usability of the new features of HMIs like proactivity and adaptvivity, in this thesis the definition of usability was extended from ISO 9241-11 standard (SFS-EN ISO 9241-11 1998). The extended definition, which has been applied is: *Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use* ***which is transparent to the user and the system***. Thus plus to the standard usability attributes (efficiency, effectiveness, satisfaction) it was introduced *transparency* to evaluate mobile proactive HMIs. Transparency can be seen as: *interpretation of the context should be coherent between the user and the HMI.*

It was shown with usability testing measurements that new features of the HMIs, which were introduced to support proactivity (like extensive visualization and multimodality), may increase the usability in terms of effectiveness, but they may challenge efficiency, while subjective satisfaction increase is not much informative. The attribute of transparency helps to correct the picture of usability. Thus the hypothesis for necessity of methodological improvements for usability testing of proactive HMIs was proved. The new usability definition and a new usability attribute are proposed for evaluation of usability issues related to proactivity. The earlier existing usability testing methodology was modified with additional questions in two points: during implementations of look-and-feel tasks the participants were asked about meaning of elements they see and tasks they expect they can solve with them; additional questions were integrated to post-test questionnaires. The methodology was tested with different devices, smartphones, tablets and laptop computers. This was done to show that the methodology works with different devices. There was no difference in a test with the different devices.

Usability testing was done in a laboratory setting. The context was brought to the laboratory setting as it is easier to control the context in laboratory setting than in a field test. This helped to get better results from the tests. Additionally, laboratory testing is less expensive and less time consuming than field testing.

The usability testing methodology proposed in this thesis was divided into two stages: α and β stage. The α stage focused on typical usability issues and gathering information about how the user understands the HMI and the context of use. After the α stage, the discovered usability issues were reported and taken care of by the development team. In the β stage, tests focused on proactive functionality of the HMI. This was done by having two different versions of the HMI: static and proactive version. In the static version, all the proactive functionality was removed. The participants did the same tasks in the usability test with both version of the HMI and the results of the tasks were compared.

The results of the use cases show that the proactive functionality increased the usability of the HMI. The HMI was more efficient to use and many of the participants that took part in the tests stated that the proactive version was better and easier to use. The results also show that the ISO 9241-11 evaluation criteria are not enough to capture proactive functionality usability.

In this thesis two use cases were presented for automation domains. The use cases were HMIs developed for building management (BM) and production management (PM) domains. The results are also considered to apply for other domains where automation systems are used.

Lessons learned

During the course of conducting the usability tests, it was discovered that, when testing with two different versions of the HMI, the development on both of the versions need to be started early on. It might be difficult to take away all the proactive functionality to have a static version of the HMI. The developers of the HMI need to be informed as early as possible what kind of versions of the HMI are needed for the usability test.

It is seen that better measures for gathering information about transparency could have been created in the methodology. In the proposed methodology, information about transparency was gathered with tasks, interview questions, and satisfacation and SUS questionnaire questions. There was no specific questionnaire targeting transparency.

Limitations

The usability testing methodology was tested with a set of mobile devices. In the BM use case, the devices used were Lenovo Thinkpad T410 laptop running windows 7 and Samsung Galaxy S2 running Android version 4.1.2 “Jelly Bean”. In the PM use case, the device used was Samsung Galaxy Tab 2 10.1 running Android version 4.0.1 “Ice Cream Sandwich”. The results are expected to apply for other similar devices.

The satisfaction questionnaire questions were slightly different for the different use cases. The satisfaction questionnaire could be refined to have the same questions for every test, no matter of the domain. This way, the results from the questionnaire could be compared across all tests which use the questionnaire.

The usability tests were conducted by inexperienced usability practitioner. The qualitative data was gathered with the best of the practitioner’s abilities. Analysis of the qualitative data could have been more profound if more qualitative data would have been gathered. For more experienced usability practitioner more qualitative data could have been gathered.

Future work

The satisfaction questionnaire could be standardized and the same questionnaire could then be used for all the tests. This would give the option to compare results related to satisfaction across different tests. Additionally, a questionnaire could be developed to gather information about transparency. The questionnaire could focus on proactive features of the HMI and how the users feel about them. With a specific questionnaire for transparency, transparency could be quantified easily.

A design pattern could be created to develop two versions, static and proactive, of an HMI simultaneously. With the use of the design pattern, it would be easy for the developers to develop two versions of an HMI for the usability test. The developers could start working on the different versions already in the beginning of the development phase. The design pattern could reduce time and cost on the development.

REFERENCES

Abras, C., Maloney-krichmar, D. & Preece, J. 2004. User-Centered Design. Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications 37, 4, pp. 445-456.

Barnum, C.M. 2011. Usability Testing Essentials: Ready, Set...Test!. Waltham, MA 024151, USA, Morgan Kaufmann Publishers. 405 p.

Berwanger, P. 2013. Alarm Management Bootcamp: Practical Deployment [WWW]. [Accessed on September 18, 2013]. Available at: http://www.isa.org/Template.cfm?Section=event_calendar1&template=/conference/ShortDescription.cfm&ConferenceID=5774.

De Boeck, J., Verpoorten, K., Luyten, K. & Coninx, K. 2007. A Comparison between Decision Trees and Markov Models to Support Proactive Interfaces. 18th International Conference on Database and Expert Systems Applications (DEXA 2007). pp. 94–98.

Brooke, J. 1996. SUS: A “quick and dirty” usability scale. Usability evaluation in industry, pp.189–194.

CamStudio.org [WWW]. [Accessed on October 3, 2013]. Available at: http://camstudio.org/.

Chen, Z., Liu, S. & Wang, X. 2008. Application of Context-aware Computing in Manufacturing Execution System. IEEE International Conference on Automation and Logistics 2008. pp. 1969–1973.

Smartmob Development 2013 [WWW]. [Accessed on October 3, 2013]. Available at: https://play.google.com/store/apps/details?id=com.andrwq.recorder.

Google, Android Design 2013 [WWW]. [Accessed on June 17, 2013]. Available at: http://developer.android.com/design/index.html.

Google, Google Glass 2013 [WWW]. [Accessed on October 4, 2013]. Available at: http://www.google.com/glass/start/.

Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., Altmann, J. & Retschitzegger, W. 2002. Context-Awareness on Mobile Devices - the Hydrogen Approach. System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference 43, 7236. 10 p.

Hong, J., Suh, E. H., Kim, J. & Kim, S. 2009. Context-aware system for proactive personalized service based on context history. Expert Systems with Applications 36, 4, pp.7448–7457.

Inico Technologies Ltd 2013 [WWW]. [Accessed on April 9, 2013]. Available at: http://www.inicotech.com/s1000_overview.html.

Inostroza, R., Rusu, C., Roncagliolo, S., Jiménez, C. & Rusu, V. 2012. Usability Heuristics for Touchscreen-based Mobile Devices. 2012 Ninth International Conference on Information Technology - New Generations, pp. 662–667.

Kaikkonen, A., Kallio, T., Kekäläinen, A., Kankainen, A. & Cankar, M. 2005. Usability Testing of Mobile Applications : A Comparison between Laboratory and Field Testing. Journal of Usability Studies 1, 1, pp.4–16.

Kangas, E. & Kinnunen, T. 2005. Applying User-Centered Design to Mobile Application Development. Communications of the ACM 48, 7, pp.55–59.

Kjeldskov, J., Skov, M. B., Als, B. S. & Høegh, R. T. 2004. Is It Worth the Hassle? Exploring the Added Value of Evaluating the Usability of Context-Aware Mobile Systems in the Field. Mobile Human-Computer Interaction-MobileHCI 2004, pp.61–73.

Liu, J., Lim, K. W., Ho, W. K., Tan, K. C., Srinivasan, R. & Tay, A. 2003. The Intelligent Alarm Management System. Software, IEEE, 20, 2, pp.66–71.

Lodhi, A. 2010. Usability Heuristics as an assessment parameter: For performing Usability Testing. 2010 2nd International Conference on Software Technology and Engineering Vol. 2, pp.256–259.

MacKenzie, I.S. 2013. Human-Computer Interaction: An Empirical Research Perspective. Waltham, MA 024151, USA, Morgan Kaufmann Publishers. 370 p.

Nayebi, F., Desharnais, J. M. & Abran, A. 2012. The state of the art of mobile application usability evaluation. Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference, pp.1–4.

Nielsen, J. 1993. Usability Engineering. San Francisco (CA): Academic Press. 362 p.

Nieto Lee, A.N., Evchina, Y., Dvoryanchikova, A. & Martinez Lastra, J.L. 2013. Proactive Content Managing System for Efficient Human Machine Interaction in Data Intensive Environments.

Petrie, H. & Bevan, N. 2009. The evaluation of accessibility , usability and user experience. The Universal Access Handbook, C Stepanidis (ed), CRC Press, p. 30.

Pope, C., Ziebland, S. & Mays, N. 2000. Analysing qualitative data. Bmj 320, 7227, pp. 114-116.

Quesenbery, W. 2001. What Does Usability Mean : Looking Beyond “Ease of Use.” Proceedings of the 48th Annual Conference, Society for Technical Communication Vol. 48, pp. 432-436.

Rubin, J. & Chisnell, D. 2008. Handbook of usability testing: How to Plan, Design, and Conduct Effective Tests, Second Edition. Indianapolis, Indiana, USA, John Wiley & Sons. 348 p.

Samsung 2013 [WWW]. [Accessed on October 4, 2013]. Available at: http://www.samsung.com/.

Sauro, J. 2011. Measuring Usability With The System Usability Scale (SUS) [WWW]. [Accessed on May 27, 2013]. Available at: http://www.measuringusability.com/sus.php.

Sauro, J. & Lewis, J.R. 2012. Quantifying User Research: Practical Statistics for User Research. Waltham, MA 024151, USA, Morgan Kaufmann Publishers. 313 p.

SFS-EN ISO 9241-11. 1998. Guidance on usability. Helsinki, Finnish Standards Association. 22 p.

SFS-EN ISO 9241-210. 2010. Human-centred design for interactive systems. Helsinki, Finnish Standards Association. 32 p.

Smith, D., 2012. Coordinating Fault Detection, Alarm Management, and Energy Efficiency in a Large Corporate Campus. ACEEE Summer Study on Energy Efficiency in Buildings 2012, pp.83–93.

Tennenhouse, D. 2000. Proactive computing. Communications of the ACM, 43,5, pp.43–50.

Vico, D.G., Woerndl, W. & Bader, R. 2011. A Study on Proactive Delivery of Restaurant Recommendations for Android Smartphones. ACM RecSys Workshop on Personalization in Mobile Applications, Chicago, USA.

Wisniewski, J. 2011. Mobile Usability. Bulletin of the American Society for Information Science and Technology 38,1, pp.30–32.

Yılmaz, Ö. & Erdur, R.C. 2012. iConAwa – An intelligent context-aware system. Expert Systems with Applications 39,3, pp.2907–2918.

APPENDIX

A. SUS questionnaire

Table A-1: *SUS questionnaire questions.*

Number	Question
1.	I think that I would like to use this system frequently.
2.	I found the system unnecessarily complex.
3.	I thought the system was easy to use.
4.	I think that I would need the support of a technical person to be able to use this system.
5.	I found the various functions in this system were well integrated.
6.	I thought there was too much inconsistency in this system.
7.	I would imagine that most people would learn to use this system very quickly.
8.	I found the system very cumbersome to use.
9.	I felt very confident using the system.
10.	I needed to learn a lot of things before I could get going with this system.

Figure A-1: *5-point Likert scale which is used in SUS questionnaires.*

When scoring SUS questionnaire the following points need to be made:

- For odd questions: subsract one from the user response
- For even-numbered items: substract the user responses from 5
- This scales all values from 0 to 4 (with four being the most positive response)

- Add up the converted responses for each user and multiply that total by 2.5. This converts the range of possible values from 0 to 100 instead of from 0 to 40.
- Now the score can be analysed and compared to results from other studies.

Since it is a standard questionnaire there are many professional tools for calculating the SUS score and for the analysis.

A good way to interpret SUS score is to convert it to percentile rank. Even though SUS score range between 0 to 100 it is not a percentage. In (Sauro 2011) average in 500 different studies for the SUS score was 68. SUS score can be converted to percentages with a process called normalizing. This process converts the score so that we can call the average score 50%. Confidence interval should calculated for the SUS score as well (see appendix B). There are software tools developed for automatic SUS score analysis and for example one can be found from (Sauro 2011).

B. Confidence interval

Three things affect the confidence interval in usability testing:

- Confidence level – Typically set to 95% or 90% in usability testing. A confidence level of 95% means that if a test was done 100 times, in 95 of the tests the test results would fall within the confidence interval.
- Variability – The more variability in the population the more wider confidence interval
- Sample size – The smaller the sample size the wider the confidence interval. The confidence interval has an inverse square root relationship to sample size.

For usability testing Adjusted-Wald interval is typically used to calculate confidence interval. The Adjusted-Wald formula is

$$\hat{p}_{adj} \pm z_{(1-\frac{\alpha}{2})}\sqrt{\frac{\hat{p}_{adj}(1-\hat{p}_{adj})}{n_{adj}}} \tag{1}$$

where $\hat{p}_{adj}$ and n_{adj} are

$$\hat{p}_{adj} = \frac{x+\frac{z^2}{2}}{n+z^2} \tag{2}$$

$$n_{adj} = n + z^2 \tag{3}$$

where *x* is the number who successfully completed the task, *n* is the number who attempted the task (the sample size) and *z* is derived from the critical value of the normal distribution for confidence level interval. Typical values in usability testing for *z* are

- 1.96 with confidence level of 95%,
- 1.64 with confidence level of 90%,
- and 2.57 with confidence level of 99%.

For example with the most used confidence level of 95% the formula basically adds two success and two failures to the results. (Sauro & Lewis 2012)